

WELCOME TO

NOV/DEC85

HAWK APPLE NEWSLETTER

HAWKESBURY
APPLE
USERS
GROUP

NO7

\$1.00

*

**DON'T
FORGET THE
NEXT MEETING
MON 25TH NOV 85
7.00PM AT
RICHMOND PRIMARY
SCHOOL**

* Cost of this newsletter to members is absorbed by group funds.

DISCLAIMER.

The H.A.U.G. Newsletter is the official newsletter of the Hawkesbury Apple Users Group, and whilst every effort is made to ensure the accuracy of the information contained therein, be it of a general, technical, or programming nature, no responsibility can be accepted HAUG as a result of the application of such information.

SOFTWARE LIBRARY SERVICE.

HAUG operates a public domain software library, containing programs written by HAUG members or provided from other user groups as well as miscellaneous public domain sources. A selection of programs is made available at general meetings for the production/media cost fee. See the HAUG LIBRARY NEWS article for more details.

YOUR COMMITTEE.

PRESIDENT:

Geoff Horsley 73 1816

VICE PRESIDENT:

Neil Cant

SECRETARY/EDITOR:

Steve Bennett 78 215

TREASURER:

Ian Lister 78 390

LIBRARIAN:

Geoff Crapp

PUBLIC RELATIONS:

Bruce Brown

CONTRIBUTORS FOR THIS ISSUE.

Thankyou to:

Steve Bennett	(Article, Programs)
Ian Lister	(Article, Programs)
Geoff Horsley	(Article)
David Zammit	(Program)
Apple User Group Sydney	(Article, Program)

SECRETARY'S EDITORIAL.

The next HAUG meeting will be the last for this year, the first meeting next year will be on the 27 JAN 86. This is also the last newsletter for 1985 the next newsletter will be after the annual general meeting which will be held on Monday the 24th of February 1986. A reminder will be sent to all members in early February. All members are invited to attend this meeting so that group activities for 1986, and services provided by HAUG to members may be improved to better suit your needs. If you don't think you got what you expected from the group, the annual general meeting will be your chance to say something about it.

I wish you all a very Merry Christmas and Happy New Year (a modem for Christmas I wonder?).

I would like to thank Ian Lister for his efforts in assisting me in producing the HAUG newsletter as without them many editions would not have been possible. See you at the meeting.

Steve B.

**** FOR SALE ****

Apple Compatible Computer with:

280 CARD
64K CARD IN SLOT 0 (112K TOTAL)
COLOUR CARD
ROM CARD
BMC GREEN SCREEN MONITOR
DRIVE CARD AND ONE SLIMLINE DISK DRIVE and
60 DISKS WITH LARGE DISK BOX.

This computer will run C/PM using programs such as Wordstar and Dbase II etc. The extra 64k of memory gives this machine lots of potential for the more serious user. The other features also make it an ideal beginners machine (colour card for games etc). The asking price is \$800 but any offers will be considered, please contact :

BOB WYNNE B/H 045 783919
 A/H 045 703367



SOFTWARE LIBRARY NEWS

The following is a brief description of the programs on HAUG library disk No 22 . I personally use the menu writer program on many of my own disks, the create program may be usefull to all the basic programmers in the group.

The programs described here are only one side of the disk and there are many other smaller progams not mentioned .The disk is available to group members for a small service charge plus the cost of a disk.

THE FIRST 6 PROGRAMS ARE BY FERG. BRAND

1:MENU WRITER A PROGRAM TO REDUCE THE NUMBER OF TITLES DISPLAYED TO THE MINIMUM. THE HELLO PROGRAM WAS CREATED BY THIS UTILITY.
READ M.WIDDESCRIPTION VIA TEXTREAD OR TEXT.

2:BYTESTRING SEARCH A UTILITY TO LOCATE BYTESTRINGS IN MEMORY.
READ BSIDDESCRIPTION FOR DETAILS.

3:DISKTRANSLATE A UTILITY WHICH EMPLOYS TECHNIQUES DESCRIBED IN 'BENEATH APPLE DOS'. USES DTIBIN.
READ DTIDDESCRIPTION FOR DETAILS.

4:HRCG DEMO A UTILITY TO ADD TO THE DOS TOOLKIT.

5:MUSICMAKER A UTILITY WHICH ALLOWS YOU TO CREATE TUNES VIA THE KEYBOARD, AND MODIFY AND/OR SAVE THE RESULT. USES MMIKEYBOARD, MMIKEYBOARD2 AND MMISOUND. SCALEIS A SAMPLE TUNE.
READ MMIDDESCRIPTION FOR DETAILS.

6:SHAPER A UTILITY FOR GENERATING HIRES SHAPES USING THE LORES SCREEN.
READ SHIDDESCRIPTION FOR DETAILS.

7:CREATE A PROGRAM DEVELOPING UTILITY, ADAPTED FROM AN ARTICLE IN 'BYTE', DEC, 1981, BY G. CRAMOND.
INSTRUCTIONS APPEAR IN THE PROGRAM BUT THE BYTE ARTICLE HELPS A LOT.

8:EVERY HIRES COLOUR 255 COLOURS FOR THOSE WITH COLOUR DISPLAYS.

10:LISTER.NC A SUBROUTINE WHICH CAN APPENDED TO AN APPLESOFT PROGRAM TO PRODUCE A STRUCTURED LISTING.

11:PICK-A-BASE A UTILITY TO CONVERT FROM ONE BASE TO ANOTHER. BECOMES UNTIDY AFTER SCREEN IS FULL.

12:BOOK REVIEW ANDREW CUNNINGHAM'S VIEWS OF JOHN F. WAKERLEY'S BOOK 'COMPUTER ARCHITECTURE AND PROGRAMMING', JOHN WILEY & SONS, 1981.

13:TEXT.READ.V4 A TEXT FILE READER WITH FRUITFUL COL. CHNL

14:BETTER MENU OK FOR CATALOGS OCCUPYING ONLY ONE SCREENFUL.

15:BINARY FILE FINDER GIVES THE START ADDRESS, LENGTH AND END ADDRESS FOR LAST BINARY FILE LOADED. USE 'CALL 768' TO RUN. WON'T WORK IF BINARY FILE LOADS INTO \$300-\$346 SPACE, WHICH IS USED BY THIS PGM.

16:CODE-TO-POKES CONVERTS A RAM RESIDENT BINARY FILE INTO A SERIES OF BASIC POKES, AND STORES THE BASIC PROGRAM ON DISK.

17:DOUBLECAT PRODUCES A TWO-PER-LINE CATALOG LISTING OF YOUR DISK. LONG NAMES ARE ABBREVIATED.

Steve B.

9:HIRESPLOT.6 A PLOTTING UTILITY FROM ANDREW CUNNINGHAM. SAMPLE DATA IS IN FILE TEST.DAT. ASSOCIATED FILES ARE HIRESPLOT.LISTING AND PLOTFILE.EDITOR.
READ HIRESPLOT.DOC FOR DETAILS.

AUG#22



Junior Filer

A feature from "Classroom Computing,
by Ashton Scholastic.

Databases are becoming a more important part of everyday life. They are used by police, governments, schools, libraries - wherever there is the need to store and have ready access to large amounts of information. This makes it extremely important that children have an understanding of how databases work and how to use them.

Junior Filer is a database that has been designed for beginners. It is easy to operate, and yet it offers you most of the search features found on commercial databases. It allows up to one hundred records to be kept, which is ample for most databases created for classroom use.

When starting out with *Junior Filer*, the user enters his/her name. When this is done the main menu is displayed. You can then select to work with a *Junior Filer* file that already exists, make a new *Junior Filer* file or end the program.

A file name must be entered for the file to be loaded or created. By entering a question mark (?), any files that are already on the disk will be displayed. When saved, *Junior Filer* files end with the suffix ".JF", but this appears automatically and does not have to be added to the filename by the user. If you have selected to work with an existing file, then it will be loaded and the file menu will appear.

If you select to create a new *Junior Filer* file, a page format must be made. Fieldnames should be entered for the categories that information is to be stored under. Up to five categories may be used. When you are satisfied with the page format, it is saved to disk and the file menu appears.

The file menu displays what you can do with the present file. A message in the upper right hand corner of the screen displays if the file is empty or full (0 pages or 100 pages). From the file menu, you may select to enter a new page, search/order/change existing pages, look at all

the existing pages, save the file onto disk or return to the main menu.

Using *Junior Filer* allows children to create and manage databases. This use of the computer fits in with work across the curriculum and with many of the topics usually studied in the classroom.

A good introduction is to have your children create a class database where they enter information about themselves. Categories can be used such as name, age, sex, height, weight, hair colour, and eye colour. The children must first collect the appropriate data, such as weight and height. This information can then be entered into the database. A printout can be made to allow the children to edit any entries that may be incorrect.

When the database has been created and is accurate, then it can be used to find out information about the class, such as how many children are older than a given age or how many weigh a certain amount. Graphs and reports can be made using the information found.

A good activity to follow this is to select a class mystery person. Using your database, a deduction game can be created to guess the mystery person. Clues are given to the class over a number of days such as "This person has brown hair", "This person is over 115 cm tall". Children use the database to gradually who it is.

Databases could be created for any topic where there are many facts involved. Some units of work where databases could be created by a class include Children's Books, Wild Flowers, Bushrangers, Cars, and Sports. These are only a few ideas. There are literally thousands of subjects which could be used for databases.

Creating and managing a database involves using many skills - gathering and sorting information, entering and editing the information, problem solving, logical thought, research skills, graphing and reporting. Many of these skills involve activities that are done off the computer. This allows a whole class to be involved in a unit of work using only one computer.

```

10 DIM RES(100,5):DS = CHR$(4): GOTO 340
20 REM ORDER PAGES
30 GOSUB 200: VTAB(20): PRINT : PRINT "Press:
'A' for Ascending (1-10,A-Z)": PRINT : PRINT TAB(
4);"or, 'D' for Descending (10-1,Z-A)"
40 VTAB(I * 3 + 1): HTAB(X + 1):AO = 0:DO = 0:
GET AS: IF AS = CHR$(27) THEN 970
50 IF AS = "A" OR AS = "a" THEN AO = 1: GOTO 80
60 IF AS = "D" OR AS = "d" THEN DO = 1: GOTO 80
70 GOTO 40
80 PRINT " "; FLASH: PRINT "SORTING INTO
ORDER.": NORMAL
90 L = 0: FOR M = 1 TO CO - 1: X = VAL(RES(M +
1,I)): Y = VAL(RES(M,I))
100 IF X = 0 AND Y = 0 THEN 140
110 IF AO AND X > Y THEN 170
120 IF DO AND X < Y THEN 170
130 GOTO 160
140 IF AO AND RES(M + 1,I) > RES(M,I) THEN
170
150 IF DO AND RES(M + 1,I) < RES(M,I) THEN
170
160 FOR N = 1 TO F: RES = RES(M + 1,N): RES(M +
1,N) = RES(M,N): RES(M,N) = RES: NEXT N: L = 1
170 NEXT : IF L = 1 THEN 90
180 PRINT CHR$(7): GOTO 710
190 REM SCREEN ROUTINES
200 VTAB(20): HTAB(1): PRINT TAB(200): RETURN
210 VTAB(3): HTAB(24): PRINT "Page# ";I; " of
";CO; " ": RETURN
220 GOSUB 240: FOR J = 1 TO F: VTAB(J * 3 + 1):
PRINT J; " "; CN$(J); " ": NEXT : RETURN
230 FOR J = 1 TO F: VTAB(J * 3 + 1): HTAB(LEN
(CN$(J)) + 5): PRINT RES(I,J): SPC(101 - LEN
(RES(I,J))): PRINT : NEXT : RETURN
240 HOME: PRINT "File ";:Z = 21 - LEN(FILES)
/ 2: HTAB(Z): PRINT FILES;: PRINT
250 PRINT
"-----": VTAB
(19): PRINT
"-----": RETURN
260 REM REMOVE PAGE
270 GOSUB 200: VTAB(20): PRINT : PRINT "Are you
sure, that you want this page": INPUT "removed?
(yes or no)";ANS$
280 IF LEFT$(ANS$,1) < "Y" AND LEFT$
(ANS$,1) < "y" THEN GOSUB 200: GOTO 1230
290 CO = CO - 1: FOR M = 1 TO CO: FOR N = 1 TO
F: RES(M,N) = RES(M + 1,N): NEXT N: NEXT M: GOSUB
200: GOTO 1300
330 REM START HERE
340 FILES = "Welcome to Junior Filer": GOSUB 240
350 VTAB(6): PRINT " Hello there .....": PRINT
: PRINT " Who am I working with today?": PRINT :
PRINT " Please tell me your first name."
360 VTAB(13): HTAB(14): PRINT " ";: HTAB(13):
INPUT " "; NAMES: NAMES = LEFT$(NAMES,12): IF
NAMES = "" THEN 360
370 P = ASC(LEFT$(NAMES,1)): IF P > 96 THEN
NAMES = CHR$(P - 32) + MID$(NAMES,2,12)
400 POKE 216,0: FILES = "Junior Filer Main Menu":
GOSUB 240
410 VTAB(6): PRINT "What would you like to do
";NAMES; "?":
420 PRINT : PRINT " 1. Make a New Junior Filer-
file."
430 PRINT : PRINT " 2. Work with an Old Junior
Filer file."
440 PRINT : PRINT " 3. Finish working with Junior
Filer."
450 PRINT : PRINT "Please press 1-3:";
460 GET AS: IF AS < "3" THEN 490
470 HOME: VTAB(11): PRINT "See you soon
";NAMES; " ": END
480 GOTO 340
490 IF AS < "1" AND AS < "2" THEN 460
500 FILES = "Enter File Name": GOSUB 240
510 VTAB(8): PRINT " Please enter the Name of
the File.": VTAB(14): PRINT TAB(5); "Enter a '?'
to Catalog the disk."
520 VTAB(11): HTAB(14): PRINT " ";: HTAB(13):
INPUT " "; FILES: IF FILES = "" THEN 400
530 IF FILES = "?" THEN HOME: PRINT
DS;"CATALOG": PRINT TAB(8); "Press a key to
continue.": GET BS: GOTO 500
540 FILES = LEFT$(FILES,12): IF AS = "2" THEN
640

```

```

550 GOSUB 240: VTAB(20): PRINT "Please enter the
Category Names that": PRINT "you want information
to be stored under.": PRINT "Press 'Return' with no
name to Finish.": PRINT "Enter a '-' to re do a
Category Name.":
560 HTAB(1): F = 0: CO = 0: FOR I = 1 TO 5
570 VTAB(I * 3): PRINT "Enter Category ";I;"s
Name.": INPUT " "; CN$(I): CN$(I) = LEFT$
(CN$(I),12): IF CN$(I) = "" THEN 600
580 IF CN$(I) = "-" AND I > 1 THEN I = I - 1: F =
F - 1: GOTO 570
590 VTAB(I * 3 + 1): PRINT I; " "; CN$(I); " ";
TAB(41): F = F + 1: NEXT
600 IF F = 0 THEN 550
610 GOSUB 220: VTAB(20): PRINT "Is this how you
want your Page set out": PRINT NAMES;: INPUT "?"
(Yes or No): ANS$
620 IF LEFT$(ANS$,1) < "Y" AND LEFT$
(ANS$,1) < "y" THEN 550
630 GOSUB 1420: GOTO 710
640 GOSUB 240: VTAB(9): PRINT "Loading the file
";FILES; " ":
650 PRINT DS;"OPEN";FILES; ".JF": PRINT
DS;"READ";FILES; ".JF"
670 INPUT F: FOR I = 1 TO F: INPUT CN$(I): NEXT
680 INPUT CO: IF CO = 0 THEN 700
690 FOR I = 1 TO CO: FOR J = 1 TO F: INPUT
RES(I,J): NEXT J: NEXT I
700 PRINT DS;"CLOSE"
710 ONERR GOTO 1620: REM MENU STARTS HERE
720 GOSUB 240: IF CO = 100 THEN VTAB(1): HTAB
(32): PRINT "File Full"
730 IF CO = 0 THEN VTAB(1): HTAB(31): PRINT
"File Empty"
740 VTAB(4): PRINT "What would you like to do
";NAMES; "?":
750 PRINT : PRINT " 1. Add a Page.": PRINT :
PRINT " 2. Search/Order/Change Pages.": PRINT :
PRINT " 3. Look at all Pages."
760 PRINT : PRINT " 4. Save File to disk.": PRINT
: PRINT " 5. Return to the Main Menu.": PRINT :
PRINT "Please press 1-5:";
780 GET AS: IF AS = "5" THEN 850
790 IF AS = "4" THEN GOSUB 1400: GOTO 710
800 IF AS = "1" THEN 900
810 IF CO = 0 THEN PRINT CHR$(7);: GOTO 780
820 IF AS = "2" THEN 970
830 IF AS = "3" THEN 1330
840 GOTO 780
850 GOSUB 240: VTAB(20): PRINT "Remember to Save
your work ";NAMES; " ": VTAB(9): PRINT "Do you
want to save your work? (Yes or)": INPUT
"No):";ANS$
860 IF ANS$ = "" THEN 710
870 IF LEFT$(ANS$,1) = "N" OR LEFT$(ANS$,1) =
"n" THEN 400
880 GOSUB 1400: GOTO 400
890 REM ADD A RECORD
900 IF CO = 100 THEN PRINT CHR$(7);: GOTO 780
910 CO = CO + 1: I = CO: GOSUB 220: GOSUB 210
920 VTAB(21): PRINT "Enter your information for
each category": PRINT "and press 'Return'. If no
information is": PRINT "to be entered then press
'Return'."
930 FOR I = 1 TO F: VTAB(I * 3 + 1): HTAB(LEN
(CN$(I)) + 4): INPUT " "; RES(CO,I): RES(CO,I) =
LEFT$(RES(CO,I),100): NEXT
940 FOR I = 1 TO F: IF RES(CO,I) < "" THEN
710
950 NEXT : GOTO 930
960 REM SEARCH PAGES INTRO
970 I = 0: GOSUB 220: VTAB(20): PRINT " Press
'Return' to move cursor, and.": PRINT : PRINT " =
equals, < less than, > greater than.": PRINT :
PRINT " @ contains, or * to put in order.":
980 EQ = 0: LTH = 0: GTH = 0: CN = 0: I = 1
990 VTAB(I * 3 + 1): X = LEN(CN$(I)) + 4: HTAB
(X): GET AS
1000 IF AS = "" OR AS = "6" THEN PRINT " ";:
GOTO 30
1010 IF AS = "=" THEN EQ = 1: GOTO 1090
1020 IF AS = "<" OR AS = "<" THEN AS = "<": LTH =
1: GOTO 1090
1030 IF AS = ">" OR AS = ">" THEN AS = ">": GTH =
1: GOTO 1090
1040 IF AS = "@" OR AS = "*" THEN AS = "@": CN =
1: GOTO 1090

```

```

1050 IF A$ = CHR$(27) THEN 710
1060 IF A$ = CHR$(13) THEN I = I + 1: IF I > F
THEN I = 1
1070 GOTO 990
1080 REM SEARCH ROUTINE
1090 PRINT A$;: GOSUB 200: VTAB(20): PRINT:
PRINT "Please enter what you would like to": PRINT
"search for and then press 'Return'."
1100 L = I: VTAB(L * 3 + 1): HTAB(X + 1): INPUT
" ";SE$:SE$ = LEFT$(SE$,25): IF SE$ = "" THEN
970
1110 I = 0: GOSUB 220: VTAB(1): PRINT CN$(L);"
";A$;" ";SE$; TAB(40): I = 1
1120 GOSUB 210: X = VAL(RES(I,L)): Y = VAL
(SE$)
1130 IF EQ AND RES(I,L) = SE$ THEN 1230
1140 IF NOT CN AND Y < > 0 THEN 1200
1150 IF LTH AND RES(I,L) < SE$ THEN 1230
1160 IF GTH AND RES(I,L) > SE$ THEN 1230
1170 IF NOT CN THEN 1290
1180 X = LEN(SE$): FOR K = 1 TO LEN(RES(I,L)):
IF MID$(RES(I,L),K,X) = SE$ THEN 1230
1190 NEXT K: GOTO 1290
1200 IF LTH AND X < Y THEN 1230
1210 IF GTH AND X > Y THEN 1230
1220 GOTO 1290
1230 GOSUB 230: VTAB(20): PRINT "Press: 'Return'
to Continue....": PRINT: PRINT " 'R' to Remove.
'C' to Change.": PRINT: PRINT " 'p' to Print.
'Esc' to cancel.";
1240 GET A$: IF A$ = CHR$(27) THEN 710
1250 IF A$ = "R" OR A$ = "r" THEN 1270
1260 IF A$ = "C" OR A$ = "c" THEN 1270
1270 IF A$ = "p" OR A$ = "P" THEN 1480
1280 IF A$ < > CHR$(13) THEN 1240
1290 I = I + 1
1300 IF I < = CO THEN 1120
1310 PRINT CHR$(7): GOTO 710
1320 REM PRINT ALL
1330 GOSUB 220: FOR I = 1 TO CO: GOSUB 210: GOSUB
230: VTAB(20): HTAB(1): PRINT "Press: 'Return'
to continue....": PRINT: PRINT TAB(9): "F" to
Flick through pages.": PRINT: PRINT TAB(
9): "Esc" to cancel.";
1340 IF A$ = "F" OR A$ = "f" THEN FOR M = 1 TO
CO: NEXT: IF PEEK(-16384) < 128 THEN 1380
1350 GET A$: IF A$ = "F" OR A$ = "f" THEN 1340
1360 IF A$ = CHR$(27) THEN 710
1370 IF A$ < > CHR$(13) THEN 1350
1380 NEXT: PRINT CHR$(7): GOTO 710

```

```

1390 REM SAVE
1400 GOSUB 240: VTAB(9): PRINT "Carefully Saving
the file ";FILES$;"."
1410 PRINT D$;"UNLOCK";FILES$;"JF"
1420 PRINT D$;"OPEN";FILES$;"JF": PRINT
D$;"WRITE";FILES$;"JF"
1430 PRINT F: FOR I = 1 TO F: PRINT CN$(I): NEXT
1440 PRINT CO: IF CO = 0 THEN 1460
1450 FOR I = 1 TO CO: FOR J = 1 TO F: PRINT
RES(I,J): NEXT J: NEXT I
1460 PRINT D$;"CLOSE": PRINT
D$;"LOCK";FILES$;"JF": RETURN
1470 ONERR GOTO 1620: REM PRINT ROUT.
1480 IF DTE$ = "" THEN GOSUB 200: VTAB(20):
PRINT: PRINT " Please enter the Date: : HTAB
(24): INPUT " ";DTE$:DTE$ = LEFT$(DTE$,13): IF
DTE$ = "" THEN 1480
1490 GOSUB 200: VTAB(20): PRINT: PRINT " Press
'Return' when printer is ready.";
1500 GET B$: IF B$ = CHR$(27) THEN GOSUB 200:
GOTO 1230
1510 IF B$ < > CHR$(13) THEN 1500
1520 PRINT: PRINT D$;"PR#1": PRINT "File:
";FILES$; SPC(67 - (LEN(FILES$) + LEN
(DTE$))): "Date: ";DTE$
1530 GOSUB 1550: PRINT: FOR N = 1 TO F: PRINT
N$;" ";CN$(N);": ";RES(I,N): PRINT: NEXT: GOSUB
1550
1540 PRINT D$;"PR#0": GOTO 1230
1550 FOR N = 1 TO 80: PRINT "=";: NEXT: PRINT:
RETURN
1560 REM CHANGE INFO
1570 GOSUB 200: VTAB(20): PRINT: PRINT "Enter
the information that you want": PRINT "changed in
each category and press": PRINT "'Return'. If the
information is correct": PRINT "then press
'Return'.";
1580 FOR N = 1 TO F: X = N * 3 + 1: Y = LEN
(CN$(N)) + 4: VTAB(X): HTAB(Y): INPUT "
";ANS$:ANS$ = LEFT$(ANS$,100)
1590 IF ANS$ = "-" THEN RES(I,N) = "": GOTO 1620
1600 IF ANS$ < > "" THEN RES(I,N) = ANS$
1610 VTAB(X): HTAB(Y + 1): PRINT RES(I,N); SPC(
101 - LEN(RES(I,N))): NEXT: GOSUB 200: GOTO
1230
1620 PRINT A$;"CLOSE": GOSUB 240: VTAB(10):
PRINT "Error #"; PEEK(222);" occurred. Press the
'Esc' key.": GET B$: GOTO 710

```

```

16 REM BASIC TRANSFER/MICROMODEM-II
REM FIRST RUN THIS PROGRAM AND THEN
REM ESTABLISH REMOTE CONTROL OF RECEIVING MACHINE
REM LEAVE TERMINAL MODE BY TYPING CTRL-A/ J
REM THEN TYPE <EXED BASIC PROGRAM TRANSFER
0 D$ = CHR$(4)
10 PRINT D$;"OPEN BASIC PROGRAM TRANSFER"
20 PRINT D$;"WRITE BASIC PROGRAM TRANSFER"
30 PRINT "POKE 1530,50181 FOR LONG FLOATING POINT PROGRAMS A GREATER DE
LAY MAY BE REQUIRED"
PRINT "POKE 1914,1"
PRINT "POKE 33,30"
PRINT "IN#0"
PRINT "PR#2"
PRINT "LIST"
PRINT "PR#0"
160 PRINT "IN#2"
170 PRINT "TEXT"
PRINT "POKE 1530"
PRINT "POKE 1914"
PRINT D$;"CLOSE"

```

--PROGRAM--

The following program was written and contributed by David Zammit who recently joined the group and is our youngest member.
 Drawer II is a doodling type program written in basic, some features of this program are:

White background on startup, incrementing diagonals and an automatic saving routine which varifies itself.

```

0 REM *****
15 REM * DRAWER VERSION II.0 * BY DAVID ZAMMIT
20 REM *****
25 TI$ = "DRAWER VERSION II.0"
30 TEXT : HOME : HTAB (40 - LEN (TI$)) / 2: PRINT TI$
35 GOSUB 85:V1 = 25
40 HTAB 1: INPUT "NAME OF PICTURE(DRAWING) >";A$
45 INPUT "PAINT BACKGROUND WHITE ";X$: IF X$ = "Y" THEN PA = 1
50 IF VAL (A$) < > 0 THEN VTAB 20: INVERSE : PRINT "FILE NAME ERROR"
: NORMAL :M= PEEK ( - 16336): GET XX$: RUN
55 D$ = CHR$ (4)
60 IF B$ = "Y" THEN 100
65 GOTO 100
70 REM *****
75 REM * INSTRUCTIONS !!! *
80 REM *****
85 VTAB 21: PRINT "KEYS-> U I O": HTAB 8: PRINT "J K": HTAB 8: PRINT
N M ,"
90 VTAB 21: HTAB 13: INVERSE : PRINT "*": HTAB 13: PRINT "*": HTAB 13: PRINT
"+": NORMAL : VTAB 21: HTAB 14: PRINT "DRAW S)TOP": HTAB 14: PRINT
"Q)INCREMENT(SIZE DOT)": HTAB 14: PRINT "(ESC)FINISHED PICTURE": POKE
25,20: VTAB 10
REM OF
100 REM *****
105 REM *
110 REM * START!! *
115 REM *
120 REM *****

125 HGR
130 IF PA = 1 THEN HCOLOR= 3: FOR I = 0 TO 159: HPLLOT 0,I TO 279,I: NEXT

135 X = 139:Y = 75:D = 0:I = 10
140 X1 = X:Y1 = Y
145 IF PEEK ( - 16384) > 127 THEN GET B$: POKE - 16384,0: GOTO 155
150 HCOLOR= 3: HPLLOT X1,Y1: HCOLOR= 0: HPLLOT X1,Y1: GOTO 155
155 IF B$ = CHR$ (27) THEN 310
160 IF B$ = "D" THEN D = 3
165 IF B$ = "S" THEN D = 0
170 IF B$ < > "Q" THEN 205
175 IF B$ = " " THEN HOME : VTAB 21: HTAB V1: PRINT "INC. 1, 19" GET
X$: PRINT " ": IF VAL (X$) = 1 THEN GOTO 185
180 GOSUB 155
185 I = VAL VTAB 21: HTAB PRINT
190 GOTO 145
205 IF
(X$) < 1 OF = (X$) > THEN
210 = (X$): VTAB 21: HTAB 1: PRINT
215 IF = "U" THEN = - 1: X = X - 1
220 IF = "O" THEN = - 1: X = X + 1
225 IF B$ = "N" THEN = - 1: X = X -
230 IF B$ = "I" THEN = -
235 IF B$ = "J" THEN Y = + = X +
240 IF = "M" THEN = -
J THE = -
B$ K THE =
  
```


DRAWER II Continued

```

245 IF Y < 0 THEN Y = 0
250 IF Y > 159 THEN Y = 159
255 IF X < 0 THEN X = 0
260 IF X > 279 THEN X = 279
265 HCOLOR= D
270 IF B$ = "D" OR B$ = "S" OR B$ = "Q" THEN GOTO 150
275 H$=X1,Y1 TO X,Y
280 Q = 1
285 GOTO 140
290 REM *****
295 REM * COMPLETE SAVE *
300 REM * HERE *
305 REM *****
310 ONERR GOTO 405: PRINT D$: PRINT
315 PRINT D$"BSAVE "A$,A#2000,L#2000"
320 PRINT D$"VERIFY"A$
325 TEXT : HOME : PRINT "DONE ..."
330 HGR
335 VTAB 21
340 PRINT "JUST RUN CURSOR OVER THIS
345 VTAB 22: PRINT " BLOAD "A$
350 VTAB 21: END.
355 GET Y$: IF Y$ = CHR$(13) THEN RETURN
360 Q1 = VAL(Y$): IF Q1 < 0 OR Q1 > 9 THEN 355
365 Y$ = Y$ + " ": GET Y$
370 RETURN
375 REM *****
380 REM *
385 REM * ERRORS GOTO *
390 REM * HERE *
395 REM
400 REM *****

405 TEXT : HOME : VTAB 9: HTAB 9: PRINT "*****"
410 VTAB 10: HTAB 9: PRINT " *ERROR TYPE PEEK (222)*"
415 VTAB 11: HTAB 9: PRINT "*****"
420 E

```

---INSTRUCTIONS---

- 1) Type name of picture
- 2) Type Y/N for painting the background
- 3) Keys used are:
 - Q - increment size (1-19)
 - D - draw
 - S - stop
 - ESC- finished
- 4) Curser control
 - U,I,O,J,K,N and M

PROGRAM

This program will allow binary file program transfers between computers written for a Micromodem II but should be easily converted to other modems.

```
19 REM BINARY TRANSFER/MICROMODEM II
20 D$ = CHR$(4)
30 PRINT D$"NOMON C,I,O"
GOSUB 420
INPUT "IS RECEIVING COMPUTER IN REMOTE MODE WITH EITHER BASIC INITIAL
IZED?":ANS$
PRINT
70 IF LEFT$(ANS$,1) < > "Y" THEN PRINT "TRANSFER ADANDONED": END
80 POKE 1530,60: POKE 1914,18: REM 600 MSEC WAIT AFTER CARRIAGE RETURN
AUTO LINE FEED IS ACIVATED AND THE WAIT FUNCTION + LOCAL DISPLAY ENA
BLED.
90 PRINT "STARTING ADDRESS-": INPUT "(MUST END WITH 0 OR 8)":ST$
100 REM LINES 110/170 - HEXIDECIMAL TO DECIMAL CONVERSION.
110 Z$ = "0123456789ABCDEF"
120 FOR I = LEN(ST$) TO 1 STEP - 1
130 FOR J = 1 TO LEN(Z$)
140 IF MID$(Z$,J,1) < > MID$(ST$,I,1) THEN NEXT J
150 DEC = DEC + (J - 1) * (16 ^ X)
160 X = X + 1: NEXT I
170 HB = INT(DEC / 256):LB = DEC - (HB * 256)
180 REM LINE 190 PLACES THE DECIMAL EQUIVALENTS OF THE HIGH & LOW BYTE
ADDRESS INTO THE PAGE 0 LOCATIONS USED BY THE MEMORY DUMP ROUTINE.
190 POKE 61,HB: POKE 60,LB
200 INPUT "NUMBER OF BYTES (DECIMAL) ":NB
210 PRINT : INVERSE : HTAB 6: PRINT "HITTING ANY KEY ABORTS TRANSFER": NORMAL

PRINT D$"PR #2"
240 PRINT "CALL-151"
250 PRINT : REM SENDS CARRIAGE RETURN.
260 FOR I = 1 TO INT(NB / 8) + 1
270 IF PEEK(-16384) > 127 THEN POKE -16368,0: GOTO 300
280 CALL 4113: REM CALLS MACHINE LANGUAGE ROUTINE BELOW.
290 NEXT I
300 PRINT
310 PRINT "3DOG"
320 PRINT D$"PR #0"
330 PRINT
340 POKE 1530,3: REM NORMAL 30 MSEC WAIT
350 PRINT " *** ALL DONE ***"
360 PRINT : PRINT "THE SENDING COMPUTERIS NOW IN TERMINAL MODE & THE REC
EIVING COMPUTER HAS BEEN RETURNED WITH BASIC UP IN REMOTE MODE."
370 PRINT : INVERSE : HTAB 15: PRINT "HIT RETURN": NORMAL
380 PRINT D$"IN #2"
390 POKE 1914,138: REM INITIATE TEMINAL MODE/FULL-DUPLEX (USE 10 FOR
HALF-DUPLEX).
400 END
410 REM LINES 420/450 LOAD RELOCATED MEMORY DUMP ROUTINE AT $1000.
420 FOR M = 4096 TO 4147: READ D: POKE M,D: NEXT M
430 RETURN
440 DATA 4,61,166,60,32,142,253,32,64,249,160,0,169,186,76,237,253,16
5,60,133,62,165,61,133,63,165,60,41,7,208,3,32,0,16
450 DATA 169,160,32,237,253,177,60,32,218,253,32,186,252,144,232,96
460 REM THE BASIC PRGM + DUMP ROUTINE OCCUPY $800-$1040. IF THE BINARY
DATA TO BE SENT RESIDES IN THIS RANGE, IT MUST FIRST BE RELOCATED W
ITH THE MONITOR MOVE COMMAND.
```

* 25 REASONS WHY YOUR COMPUTER MAKES *
* AN IDEAL PARTNER *
* *****

- 1) A computer is always ready when you are.
- 2) A computer obeys your every command.
- 3) A computer never gets a headache.
- 4) A computer will respect you in the morning
- 5) A computer doesn't cause problems it solves them.
- 6) A computer accepts you just the way you are.
- 7) A computer listens to reason.
- 8) A computer doesn't snore.
- 9) A computer won't take you for granted.
- 10) A computer never compares you to its past lovers.
- 11) A computer doesn't take forever in the bathroom.
- 12) A computer doesn't have a father who owns a shotgun.
- 13) A computer won't take up all your wardrobe space.
- 14) A computer is a lot smarter than anyone else you've ever dated
- 15) A computer won't leave hard to remove stains on your sheets
- 16) A computer won't embarrass you in front of your parents or friends.
- 17) A computer doesn't make you feel guilty about anything.
- 18) A computer won't grab all the blankets in the middle of the night.
- 19) A computer will never ask you to spend holidays with its family
- 20) A computer won't use up all the hot water in the shower.
- 21) A computer won't get upset if you just roll over and go to sleep
- 22) A computer won't leave you if it finds out you have been unfaithful.
- 23) A computer won't make you sign a pre-nuptial agreement.
- 24) A computer can't test against you in court.
- 25) A computer can't give you aids.

